DEPARTMENT OF EDUCATION - STUDENT FINANCIAL ASSISTANCE

FINANCIAL MANAGEMENT SYSTEM

RELEASE 3.3 DESIGN TEST PLAN

Creation Date: 04/11/2001
Last Updated: 07/02/2001

Document Control

Reviewers

Name	Position
Paul Stonner	Client Project Manager
Francesca Keating	Contractor Project Manager
Barry Weiss	Implementation Manager
Amy Woodson	Transition Manager

Distribution

Copy No.	Name	Location

TABLE OF CONTENTS

1. DESIGN TESTING OVERVIEW

Design Testing (DT) is the functional test of the system prior to production, and validates the system's design through the simulation of actual business processes. Business functions and system functions (i.e. batch processing, system security features, etc.) are executed in a controlled environment using predefined test scripts to ensure integrity and confidence in the results. A Design Test also incorporates the new business procedures to validate that the system properly enables their execution. The system's business events are examined from the user or business perspective, rather than attempting to test every possible condition from the design perspective. The developer's unit testing approach allows for robust testing that exercises logical branches and conditions.

The goal and objectives of a Design Test - in addition to validating the design - are to confirm set ups, validate configurations, conversions, training material and ensure user requirements are met. The Design Test will also confirm cross-functional handoff points and processes. The testing should also identify and resolve all go-live issues. The deployment planning approach and assumptions should also be considered and captured during Design Testing.

There are many tasks, deliverables and procedures in the execution of a Design Test. This document provides the details of the activities and work products. Samples of work products are included in the exhibit section of this document. All scripts and work products will be placed under configuration management. Any changes and updates to scripts will follow CM guidelines.

Entry and Exit criteria have been established as management control gates for the design test process. Entry criteria have been captured and listed on the following checklist.

Design Test Workstream Readiness Checklists:

Prior to entry into a Design Test there are a number of activities and events that must take place and documents that should be completed. These events and documents have been captured in the DT workstream readiness checklist and will be reviewed at the readiness session by the management office. Each workstream will be required to complete all relevant sections and submit for review. A sample of this document is displayed in the exhibit section. All sections of the checklist must be signed off as being completed or have signature approval *in lieu* of compliance.

Exit Criteria for Design Testing:

Exit criteria can be summarized in the following points:

- All Design Test controlled unit and integration testing scripts and cycles have been executed successfully.
- All identified errors and defects, as well as issues and incidents, have been properly documented and worked through the resolution process.

- All reviews have been conducted and the reviews yielded satisfactory results.

2. DESIGN TEST SCOPE AND EXPECTATIONS

The FMS Phase III Design Test I scope includes testing the following program interfaces:

- Direct Loan Consolidation
  - The design test scripts will include testing of only **Non GAPS** transactions interfaced to the FMS General Ledger and also the export to the ED CFO Oracle instance. The transaction files will contain "unbooked" loan data.

- Direct Loan Origination
  - The scope of this cycle of DLO testing will test the connection points of the Loan Origination feeder system to FMS. The test scripts will test the file transfer "pass through" of Loan Origination files directly to GAPS. The file format will be in Pre-October GAPS format and will not be altered.

- Direct Loan Servicing
  - The scope for DLS will include the processing of booked loans, payments, adjustments, and refunds from IF10 transfer files. This financial data will be imported into the FMS General Ledger and verified.
  - G Records containing FAD Deposits and FIG Interagency transfers transactions will also be imported into the FMS GL in DT 1.
  - IF20 records (i.e. unapplied/misdirected payments) will also be processed.

- LEAPP/SLEAPP
  - The performance reports of the LEAPP/SLEAPP will be tested.

- FFEL Lender
  - The scope of FFEL lender will be to process collection and disbursement data up to the point of loading the transactions into the FMS instance and exporting that file out into the format for transmitting to FMSS. Integration testing of loading the file into FMSS will be tested in the next test phase within the end-to-end integration test scripts.

- Debt Collections
  - The scope of the Debt Collections Services (DCS) interface is to allow accounting transactions to be created in SFA FMS general ledger from the Collections and disbursements (Accounting) files generated by Raytheon twice a week. Included in this scope is the conversion of the old PAS CAN account structure into the current SFA Account Code Structure.

In addition, the Design Test will execute regression testing for the FFEL GA program to ensure that no effects in functionality are found due to multi-org configuration. Design Test I will also include creating the export file containing summary data of test script transactions. This file will eventually flow to OCFO FMSS.

The FMS Phase III Design Test II scope includes testing of the following program interfaces:

- PELL

- Direct Loan Origination
    - Design Test II will process "unbooked" loan transactions and excess cash into the FMS GL, and payment requests transactions into FMS AP and export to GAPS.

- Direct Loan Consolidation
    - Design Test II will incorporate testing of GAPS transactions. Obligations, Payment, De-obligations, Refunds, and Excess Cash transactions will be interfaced to FMS AP and GL and interfaced out to GAPS.

- Campus Based
    - The scope for the Campus-Based program will include processing of the following
        - Budgetary Journal Entries
        - FSEOG, FWS, and Perkins Loan award obligations (positive/negative)
        - Teacher Cancellation Payments
        - FISAP Expenditures
        - Perkins Loan Balance Sheet Data

- GAPS
    - The scope of this testing will incorporate the interface of Pell, CB, DLO, and DLC data to GAPS and the receipt of GAPS data back to FMS

- Regression Tests of all programs.

The Design Test cycles will include unit test scenarios by program and integration test scripts for cross-program business processes such as the "Close" process. Oracle's Muti-Org functionality will also be tested within unit and integration test scripts.

Unit and integration test cycles for the application development work stream must be completed prior to the start date of the Design Test. This checkpoint has been included in the readiness checklist.

Three weeks have been allocated for the execution of the Design Test I and resolution of issues. Design Test II will be scheduled for 3 additional weeks of testing. Given the external GAPS / ED CFO schedule dependencies, the exact date for Design Test II is still to be determined. A Design Test III will be scheduled, if needed, as determined by management office.

3. DT TEST SCHEDULE

Execution of Design Testing will occur from June 25, 2001 through July 13, 2001. A Test Readiness Review will be held prior to the start date of testing to verify all entry criteria have been met. The test execution phase involves executing each of the test cases and comparing the actual results with the expected results. Execution of the cases may sometimes occur more than once during the test execution phase, since discrepancies will be identified and fixed during this phase. Unit and integration test cases will be executed until no discrepancies are found or other resolutions or work-arounds are identified. Review and analysis of DT Test Results will be conducted on a daily basis. The FMS program leads will perform the acceptance testing (PAT) for the development inventory prior to code migration to the test instance. The chart below provides details on the Design Test schedule. This chart is subject to change at management discretion. All official dates will be reflected in project plan.

ID	Task Name	Start Date	End Date
	Product Test Schedule		
1	DT Planning	28-MAY	10-JUN
2	Logistics/Infrastructure	04-JUN	13-JUN
3	Design Documentation	28-MAY	14-JUN
4	Integration Test Planning	31-MAY	14- JUN
5	Unit Test Planning	31-MAY	14-JUN
6	FMS PAT of Development Inventory	11-JUN	16-JUN
7	Instance Creation and Preparation	18-JUN	22-JUN
8	Pre-DT Reviews	19-JUN	20-JUN
9	DT I Test Readiness Review	22-JUN	22-JUN
10	DT Execution	25-JUN	13-JUL
11	Design Testing Kickoff Meeting	25-JUN	25-JUN
12	DT Unit Testing	25-JUN	13-JUL
13	CFO – Participation	27-JUN	13-JUL
14	DT Integration Testing	1-JUL	13-JUL
15	DT I Closure – Exit Meeting	13-JUL	13-JUL
16	Design Test II Planning	2-JUL	18-JUL
17	DT II Instance Creation and Preparation	1-AUG	5-AUG
18	Design Test II Test Readiness Review	3-AUG	3-AUG
19	DT II Execution	6-AUG	24-AUG
20	Design Testing Kickoff Meeting	6-AUG	6-AUG
21	DT II Unit Testing	6-AUG	24-AUG

ID	Task Name	Start Date	End Date
22	CFO – Participation	13-AUG	24-AUG
23	DT II Integration Testing	13-AUG	24-AUG
24	DT II Closure – Exit Meeting	24-AUG	24-AUG

4. DT ROLES AND RESPONSIBILITIES

Staffing for DT involves both SFA FMS Implementation Team and SFA personnel. The primary roles for the Design Test effort include Test Coordinator(s), Program Functional Workstream, Application Development and Support (AD&S), Technical Infrastructure, Integration Workstream, Management office and others. Complete details of ownership have been documented in the Roles and Responsibility matrix. A summary of roles and responsibilities are described below:

4.1. TEST TEAM COORDINATOR

- Develop and maintain the Design Test plan and Design Test work products.
- Develop Design Test status reporting.
- Define issue resolution and recovery approach.
- Review test scripts inventory, scenario test cases and test data.
- Work with the program leads in coordinating test scripts to minimize cross-team testing conflicts.
- Develop, review and utilize Design Test readiness checklist to ensure needed application set up or configuration requirements are coordinated and performed.
- Coordinate all test activities for the team according to the test plan.
- Assist in evaluating and recording test results.
- Provide feedback and progress reports to management office regarding status of product test activities and significant issues.
- Assist in review of regression testing to validate changes to the application software and/or configuration. Regression testing will consist of testing the programs and affected sub-systems following the multi-org modification.

4.2. PROGRAM WORKSTREAM

- Development of the Design Unit and Integration Test Scripts.
- Execute the tests.
- Evaluate test results.
- Record all incidents and problems encountered during testing activities.
- Conduct regression testing to validate changes to the application software and/or configuration.

4.3. MANAGEMENT OFFICE

- Participate in the walkthrough of the Design Test Plan.
- Review all incidents and problems encountered during testing activities.
- Monitor the testing process to identify problems, mitigate potential risks and schedule slippage.
- Review test results with the test team to clarify questions, concerning system functionality and discrepancies with expected results, and ensure that the design testing activity was a valid and complete exercise.
- Confirm that the testing process is comprehensive in scope and is complete.
- Review and determine that Design Test Exit Criteria have been met.

4.4. Integration Workstream

The integration workstream is comprised of a cross section of program team members, management office, and AD&S.

4.5. Technical Infrastructure

The infrastructure team is responsible for the Instance Creation and preparation as well as System support during execution of the Design Test. The Design Test landscape should mirror the production landscape as much as possible with respect to architecture. All configuration management procedures should be followed for all code changes and migrations to the test environment. All changes or updates to the testing environment will require change requests under configuration management guidelines.

5. EXECUTING AND VALIDATING TEST CASES

A test case or scenario is a discrete, executable activity that will return a predictable result. The scope of the test cases/scenarios to be included will be identified through the scenario description element on the scenario test matrix or the overview section in the test script file. Test cases are made up of a setup, test description, and expected result sections (see Exhibit 1). A test case is designed to verify the functionality of a business scenario, and describes step-by-step how the business scenario is being tested.

Executing the test cases is the responsibility of the entire test team, as coordinated by the Test Team Coordinator. Any team member performing a test will update the appropriate fields of the test script matrix when conducting Design Testing.

Validation is defined to be the comparison of the actual results to the expected results. If the actual data results match the expected results then the test script passes. SFA CFO and SFA FMS personnel will be asked to initial test scripts that passed the controlled test script.

6. ISSUE TRACKING AND PROBLEM DIAGNOSIS AND RESOLUTION

The test team member conducting the test will record any incidents or problems that are related to product testing. An analysis of the incident will be performed to determine if the problem was caused by:

- Problem(s) with the test data;
- Problem(s) in the test environment;
- An incorrectly run test script;
- A misunderstanding of what the expected result should be.

If necessary, the incident will be forwarded to the business requirements, software architecture or technical support team for assistance in this analysis.

Incidents or problems will be recorded using the "Design Testing Incident Report" (see Exhibits). The "Design Testing Incident Report" is designed to capture as much information as possible to relay to the business requirements, software architecture or technical support team. This information will enable recreation of the situation. test team members completing a "Design Testing Incident Report" should:

1. Write down a description of the symptoms that occurred and the test execution steps leading up to them, noting the date and time of the discrepancy.

2. Print any screen(s) or report(s) that illustrate the error that is occurring.

3. Escalate the problem to the Test Team Coordinator immediately if this is a "show stopper" that prevents continuation of the test.

4. If the problem isn't severe, log the information and continue with the same or other tests.

A "discrepancy" is a difference between what was *expected* to happen and what *actually* happened. Discrepancies are initially evaluated to ensure that they represent system defects. The initial evaluation of discrepancies includes:

- Review of the test data;
- Review of the test environment;
- Review of test cases;
- Review of actual system functioning.

If it is determined that the "discrepancy" represents a system defect, then the incident will be forwarded to the business requirements, software architecture or technical support team for assessment and proposed solution(s). The proposed solution(s) may be technical/development or business process based (or both). Further, some solutions will entail significant technical or development work whereas others will not (and likewise for business process-based options). Once the solution options have been identified, the Test Team Coordinator will work with project management in determining which option will be followed or if the resolution will be deferred.

Standard procedures for development, unit test, code walkthrough and migration will be followed for any solution for which technical/development work is included. The solution will be tested as part of the Design Test, and regression testing as needed will be conducted for the systems and processes impacted by the modification. After the solution has been successfully tested, the incident will be designated as "closed" by the Test Team Coordinator.

7. TEST TRACKING AND DOCUMENTATION

The actual test results will be compared to the expected results to determine if the test ran correctly (or incorrectly). At the completion of each test activity the Test Team member will evaluate the status of the activity as "Pass", "Retest", "Fail" or "Blocked". A status of "Pass" indicates that the expected results were achieved. A status of "Retest" indicates that the actual results do not match the expected results for reasons that can be corrected and retested. A status of "Fail" indicates that actual results do not match expected results and the errors cannot be corrected during the Design test. A status of "Blocked" indicates that the test cannot be executed because of missing system components. Explanations must be provided for all cases with status of "Blocked". Design Testing is considered complete when each test case has a status of "Pass" or "Blocked," and the results have been reviewed by the Test Team Coordinator and the Management Office.

Each scenario or combination of scenarios will be tracked via several checkpoints consisting of key processes. Each checkpoint will be measured on a Red, Yellow, or Green Basis:

- Green – The checkpoint produced the expected results
- Yellow – The checkpoint did not produce the expected results or raised major issues, however, there is an identifiable solution that can be implemented prior to "Go Live"
- Red – The checkpoint could not be completed or did not produce the expected results for unknown reasons that cannot be resolved prior to "Go Live"

Swat teams will be created and disbanded as needed by the management office to resolve critical issues.

8. TEST DATA

Sample data will be gathered from the programs. Each program will be responsible for creating sample data files or using a subset provided by the feeder systems.

9. TEST SCRIPTS

A test script is a collection of related test cases, put into a sequence of steps. The Test Team Coordinator, the management office, and the integration test team will review all unit test and integration test scripts. Any test scripts that have been determined to have dependencies on other scripts will have the appropriate sequencing documented in the prerequisite test column. All test scripts should be placed under configuration management upon review by the Test Team Coordinator and management office.

10. TEST CYCLES

A test cycle is a collection of test scripts put into a sequence. A test cycle describes how a group of test scripts will be executed. Dependencies between scripts should be identified and recorded by FMS SFA functional program leads. The FMS PMO will be responsible for the overall management, scheduling and review of Design Test execution.

11. DESIGN TEST RESULTS

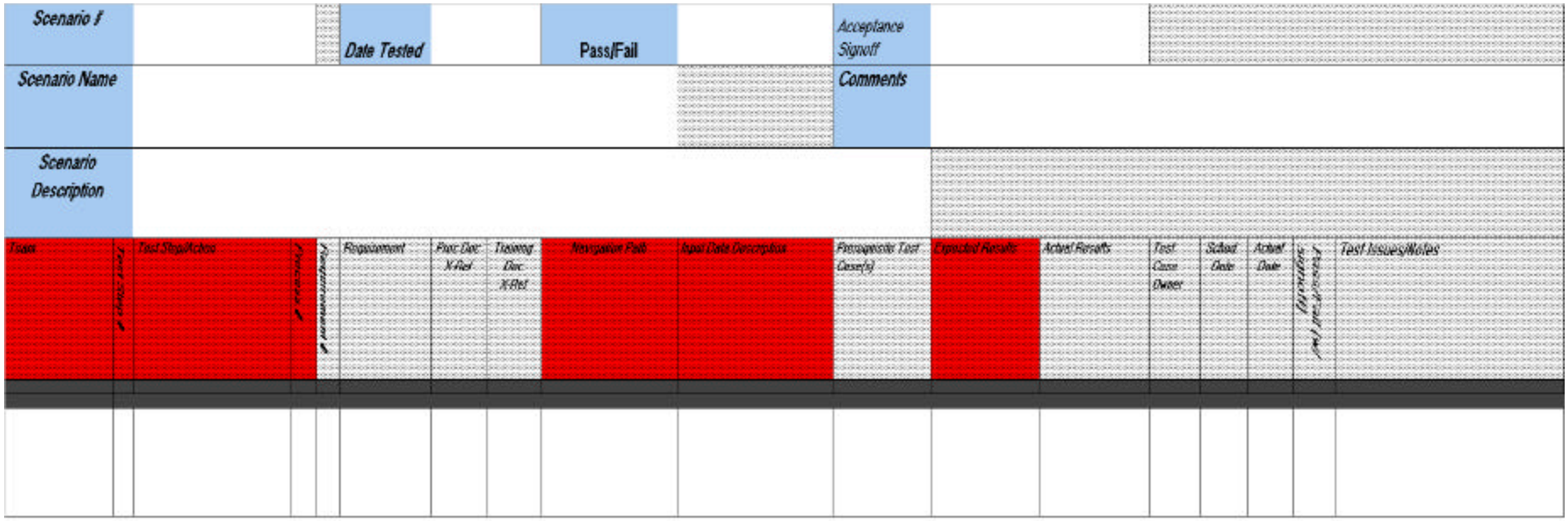
The Design Unit Test Results will be included and recorded on the Summary Status Results Matrix. All results will be summarized and presented to management office for review and submission as a Design Test Exit Criteria checkpoint.

12. CLIENT ACCEPTANCE APPROACH

The acceptance of Phase III will be based on the review and acceptance of the Design Testing results. These results provide confirmation that the transactional data was processed into the current FMS instance and interfaced to ED CFO correctly. The SFA CFO and SFA CFO project Lead will provide production release approval contingent upon this acceptance.

13. EXHIBITS

13.1. Exhibit 1 Unit and Integration Sample Test Script Template.



13.2. EXHIBIT 2 – DEFINITIONS OF TEST CASE/SCENARIO ELEMENTS

Scenario number: <Program acronym or 1st 2 letters of program> + number

Scenario name: Name of scenario. Name should reflect business process being testing.

Scenario Comments: Identify the test's general business process area.

Scenario Description: Describe briefly the objective of the test, such as to verify that something works a certain way, or to ensure that invalid data is identified, etc.

Interfaces: Identify any interfaces that are needed for the test.

Reports: Describe reports or queries to be used to help in this test or in analyzing the results of the test.

Set-up: Document any applications setup requirements. Also document any seed data requirements that might be needed for this test.

Data Sheet (Separate Tab): Document all required information of input data. Provide information such as input file names from feeder system. Include explanation of content of data. Include any infrastructure requirements for transfer/storage of data.

Sample information include:

- A sample <name or type of> interface file with ### records with no anticipated errors
- A sample <name or type of> interface file with ### records with an error in the GAPS Award ID field (specify invalid data or invalid value)
- Etc.

Test Step / Action: Describe the anticipated test procedures or steps, as detailed as is necessary to adequately complete the test.

Navigation: Include all Oracle Navigation paths and necessary parameters.

Expected Results: To be completed before the test is run. Describe the expected outcome of the test. This can be included on separate tab if more space is needed

Proc-Doc Xref: Reference to process documentation related to this step in the script if any.

Test Case Owner: Person performing the test.

Date: Date test step executed.

Actual Result: Describe whether the test resulted as expected or not; reference output documents, screen shots, etc., to provide documentation of results. This can be documented as a separate sheet.

Test Issues/Notes: List related test notes to be recorded.

Requirement #: Enter requirement from requirements matrix, if applicable.

Prerequisite Test Case(s): List other unit or integration test scripts that should be executed, if any, prior to this step.

Comments and Other Info: (as needed)

13.3. EXHIBIT 3 - TESTING INCIDENT REPORT

DESIGN TESTING INCIDENT REPORT	
INCIDENT REPORT #:	DATE:
Name:	Phone Number:
Subsystem/Functional Area(s) affected:	Test Case #:
Problem with (check all applicable): ☐ Environment ☐ Software ☐ Other _____	Activity (select one): ☐ Design Test ☐ Other _____
Severity of Problem: ☐ Severity 1 – Major problem, system doesn't respond or crashes, loss of data, feature(s) inoperative or will not execute ☐ Severity 2 – Major feature halts, incorrect results after execution, other severe restrictions ☐ Severity 3 – Expectations of major feature not met, problem is obstacle that can be worked around ☐ Severity 4 – Feature executes correctly but minor cosmetic change required to meet test objectives	
☐ Check if Recurring Problem (explain):	
Description of Problem:	
Possible Cause/Solution:	
FMS FUNCTIONAL ASSESSMENT (this activity to be performed by member of Phase III Business Requirements staff)	
Assigned To:	Date:
Assessment:	
Recommended Solution(s):	
FMS SOFTWARE ARCHITECTURE/TECHNICAL ASSESSMENT (this activity to be performed by AD&S or Technical Infrastructure staff)	
Software Architecture/Technical Assessment Performed by:	Date:
Assessment:	
Software Affected:	
Solution(s):	
CORRECTIVE ACTION	
Resolved by:	Date Completed:
Corrective Action Taken:	

TEST OF CORRECTIVE ACTION		
Date:	Results: ☐ Pass ☐ Retest ☐ Fail ☐ Blocked	Tester Signature:

13.4. Exhibit 4. Readiness Check list

Phase III Release CRP
Workstream Readiness Checklist

Workstream:

	Deliverables	Owner(s)	Reviewer(s)	Due Date	Signoff*	Signoff Date	Notes/Document References
1)	CRP Participants						
	A. Participants Identified						
	B. Participants Notified of CRP Schedule						
	C. Superuser Mentors Identified and Notified						
2)	Design Documentation						
	A. Master Process List Updated						
	B. Requirements Matrix Updated						
	C. Process Flows Completed						
	D. CE Documents Approved						
3)	Issues						
	A. Phas III Issues Resolved & Updated in SFA Tracker						
4)	Unit Test Scripts						
	A. Scripts Defined (incl. Data & Expected Results)						
	B. Dependencies Documented (intra- & inter-WS)						
	C. Testers Assigned						
	D. Data Requirements Identified						
	E. Interface & Extension Execution Requirements Identified and Communicated to Tech Team						
	F. Training Procedures & Navigations Linked to Tests						
5)	Integration Test Scripts	To Be Developed by Integrated Testing Team					
	A. Test Scenario(s) Defined						
	B. Test Flow(s) Developed						
	C. Testers Assigned						
	D. Data Requirements Identified						
6)	Stress Test Requirements	Not Applicable for this CRP					
	A. High-frequency Processes Identified						
	B. Stress Test Scripts Created						
	C. Locations, Users & Frequencies Identified						
7)	Configuration Guides						
	A. Configurations Updated						
	B. Module-specific Configurations Updated						
	C. DFF Configurations Updated						
8)	Client Acceptance Tests						
	A. Conversion CATs Completed						
	B. Interface CATs Completed						
	C. Extension CATs Completed						
	D. Reports/Queries CATs Completed						
9)	Baseline Instance						
	A. Baseline Updated for Creation of CRP Instance						
10)	Responsibilities & Menus						
	A. Custom Responsibilities & Menus Identified						
	B. CRP Participants Mapped to Responsibilities						
	C. Information Communicated to Tech Team						
11)	CRP Instance						
	A. CRP-specific Configurations Completed						
	B. Manual Data Entered						
12)	Pretesting						
	A. Unit Test Scripts Pretested						
	B. Responsibilities & Menus Pretested						
13)	Training Documents						
	A. Procedures Completed						
	B. Navigations Completed						
14)	CRP Binders						
	A. Workstream CRP Binders Completed						
	B. Superuser Binders Completed						
	C. Binders Reviewed						

Certification

Based on a review of the items listed above, we have determined that:

☐ The Workstream is ready to begin the R1W2 CRP

☐ The Worksteam can begin the R1W2 CRP with the following conditions:

☐ The Workstream is not ready to begin the R1W2 CRP due to the following reasons:

Signoffs			
	SFA FMS Business Workstream Lead		
	FMS ITS Workstream Lead		Date
	KPMG Workstream Lead		Date
			Date

*** Signoff signifies that**

1) The Reviewer considers the item to be sufficiently complete to begin the CRP for the agreed upon system functionality to be tested

2) The process designs have been reviwed and agreed to with respect to what is to be tested and how

13.5. Exhibit 5 Roles and Responsibilities

Student Financial Assistance
U.S. Department of Education

We Help Put America Through School

Item #	Task	Functional Workstream	AD&S	Tech. Infrastructure	Integration Workstream	Management Office	Other
		O=Owns, P=Participates, C=Coordinates, R=Reviews					
CRP Planning							
1	Create Master CRP Schedule & Publish	R	R	R	O	R	
2	Assign CRP Participants	O			C	R	
3	Assign TI & ADS Support Personnel		O		C	R	
4	Create CRP Participant Roster	R	R	R	O	R	
5	Plan Team Event(s)				C	R	CM Owns
6	Develop CRP Status Reporting Approach				O	R	
7	Define Issue Resolution & Recovery Approach				O	R	
8	Create CRP Kickoff Presentation				O	R	
Logistics/Infrastructure							
9	Determine Number of Rooms Required				O		
10	Identify and Schedule Rooms				O		
11	Finalize Team/Room Assignments	R	R	R	O		
12	Identify CRP TI Requirements (PCs, Ports, Printers, etc.)	O		P	C		
13	Setup & Test PCs, Ports, Printers, etc.			O	C		
Integration Test Planning							
14	Assign Integration Test Point-persons	O			C		
15	Identify Integration Test Requirements	R	R	R	O	R	
16	Develop Integration Test Scripts	P	P	P	O		
17	Determine I&E Processing Requirements	P	P	O	C		
18	Assign Integration Testing Team	O			C		
19	Publish Integration Testing Schedule	R	R	R	O	R	
Unit Testing Planning							
20	Develop Unit Test Scripts for New Functionality	O			R		
21	Identify Prior Release Unit Tests for Regression Testing and Modifiy as Necessary	O			R		
22	Create Unit Test Schedule	O			R		
23	Assign and Schedule Unit Test Executors	O			R		
Design Documentation							
24	Update Requirements Matrix	O			R		
25	Update Master Process List	O			R		
26	Complete CE Documents	O			R	R	
27	Complete Functional Specs	O			R		
28	Complete Technical Specs	P	O		R		
29	Complete Batch (Maestro) Schedule Requirements	O		P	R		
30	Deployment Site Infrastructure Reqs (PCs, Printers, etc.)	O		P	C		
31	Define Performance/Stress Requirements	P	P	P	O		
32	Update Development Inventory	P	O		C		
33	Update Configuration Guides	O			R		
34	Update Business Processes (Level 4)	O			R		
35	Update System Flows	P			O		
36	Update Data Flows	P			O		
37	Update User Responsibility Matrix	O			C		
38	Final Design Review Completed	P			O		
39	Complete Release (Features) Document	P			O	R	
Training Documentation & Planning							
40	Develop Procedures & Navs (New Functionality)	O					TR Coordinates
41	Update Procedures & Navs (Existing Functionality)	O					TR Coordinates
42	Develop Release Training Plan	R					TR Owns
CAT							
43	Complete CAT for Development Items (incl. Transfers)	O	P		C		
44	Document and Resolve CAT Issues	O	P		C	R	
45	Approve Development Items for CRP	O	O		C		
46	Move Development Items to CRP	P	O		C		

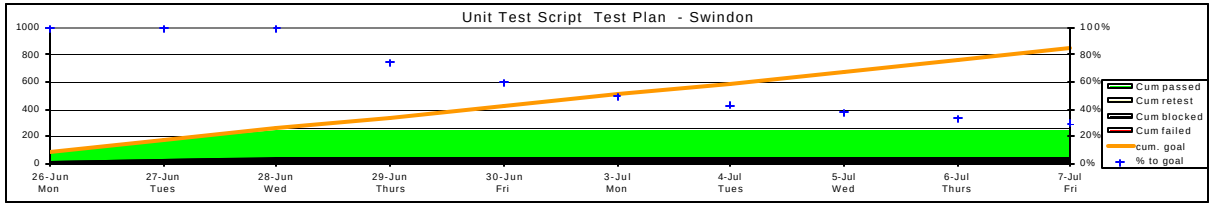
13.6 Exhibit 6 Results Matrix

Tracking Unit Tests

SPCD Unit Testing - Swindon

nter items in Blue boxes

	Total scripts to test	Total Retests Required		Mon 26-Jun	Tues 27-Jun	Wed 28-Jun	Thurs 29-Jun	Fri 30-Jun	Mon 3-Jul	Tues 4-Jul	Wed 5-Jul	Thurs 6-Jul	Fri 7-Jul	Cumulative Total
urchasing	124	1	tested	12	12	12	0	0	0	0	0	0	0	36
			goal	12	12	12	12	12	12	12	12	12	12	29%
			passed	10	11	12	0	0	0	0	0	0	0	33
			Retest	1	0	0	0	0	0	0	0	0	0	1
			blocked	0	0	0	0	0	0	0	0	0	0	0
			Failed	1	1	0	0	0	0	0	0	0	0	2
ventory	234	8	tested	23	23	23	0	0	0	0	0	0	0	69
			goal	23	23	23	23	23	23	23	23	23	23	29%
			passed	20	18	20	0	0	0	0	0	0	0	
			Retest	3	3	2	0	0	0	0	0	0	0	
			blocked	0	0	0	0	0	0	0	0	0	0	
			Failed	0	2	1	0	0	0	0	0	0	0	
IP	96	0	tested	10	10	10	0	0	0	0	0	0	0	30
			goal	10	10	10	10	10	10	10	10	10	10	31%
			passed	8	8	8	0	0	0	0	0	0	0	
			Retest	0	0	0	0	0	0	0	0	0	0	
			blocked	0	0	0	0	0	0	0	0	0	0	
			Failed	2	2	2	0	0	0	0	0	0	0	
lanning	117	4	tested	12	12	12	0	0	0	0	0	0	0	36
			goal	12	12	12	12	12	12	12	12	12	12	31%
			passed	10	11	8	0	0	0	0	0	0	0	
			Retest	2	0	2	0	0	0	0	0	0	0	
			blocked	0	0	0	0	0	0	0	0	0	0	
			Failed	0	1	2	0	0	0	0	0	0	0	
NG/BOM	153	5	tested	15	15	15	0	0	0	0	0	0	0	45
			goal	15	15	15	15	15	15	15	15	15	15	29%
			passed	15	10	12	0	0	0	0	0	0	0	
			Retest	0	4	1	0	0	0	0	0	0	0	
			blocked	0	0	0	0	0	0	0	0	0	0	
			Failed	0	1	2	0	0	0	0	0	0	0	
osting	122	8	tested	12	12	12	0	0	0	0	0	0	0	36
			goal	12	12	12	12	12	12	12	12	12	12	30%
			passed	8	10	10	0	0	0	0	0	0	0	
			Retest	4	2	2	0	0	0	0	0	0	0	
			blocked	0	0	0	0	0	0	0	0	0	0	
			Failed	0	0	0	0	0	0	0	0	0	0	
Totals	846	26	tested	84	84	84	0	0	0	0	0	0	0	252
			goal	85	85	85	85	85	85	85	85	85	85	846
			passed	71	68	70	0	0	0	0	0	0	0	209
			Retest	10	9	7	0	0	0	0	0	0	0	
			blocked	0	0	0	0	0	0	0	0	0	0	
			Failed	3	7	7	0	0	0	0	0	0	0	17



cum. tested	84	168	252	252	252	252	252	252	252	252	
cum. goal	84.6	169	254	338	423	508	592	677	761	846	846
% to goa	99%	99%	99%	74%	60%	50%	43%	37%	33%	30%	
Cum blocked	0	0	0	0	0	0	0	0	0	0	
Cum failed	3	10	17	17	17	17	17	17	17	17	
Cum passed	71	139	209	209	209	209	209	209	209	209	
Cum retest	10	19	26	26	26	26	26	26	26	26	

- Unit Tests will receive one of 4 testing statuses:
  - **PASS**—Actual results match the expected results and no significant issues were identified
  - **RETEST**—Actual results do not match the expected results for reasons that can be corrected and retested
  - **FAIL**—Actual results do not match the expected results and the errors cannot be corrected during the DT
  - **BLOCKED**—The test cannot be executed because of missing system components (e.g., interface or extension)